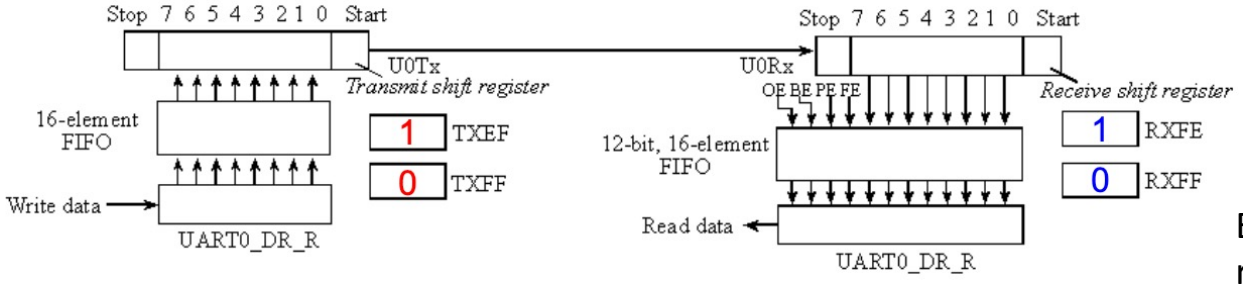
Lab 6 Introduction

UART Initialization, Programming and Interrupts

Lab 6

Example: key pressed and character sent from PuTTY on PC through its UART



Example: character received on CyBot through TM4C microcontroller UART

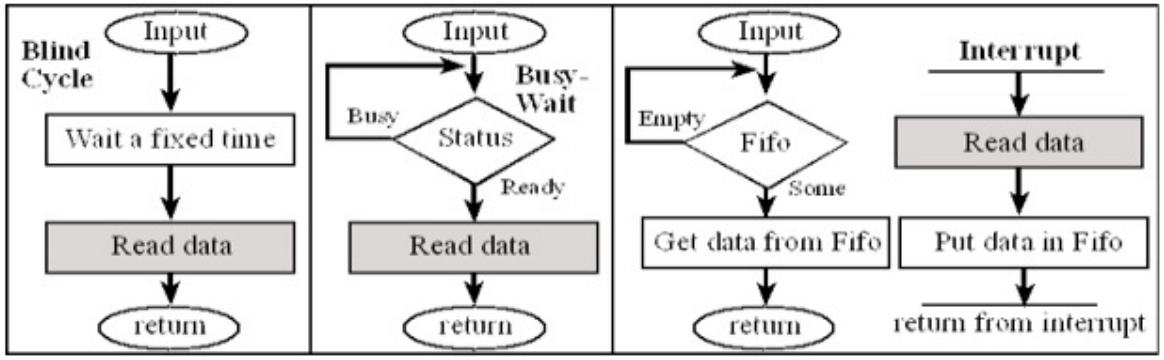
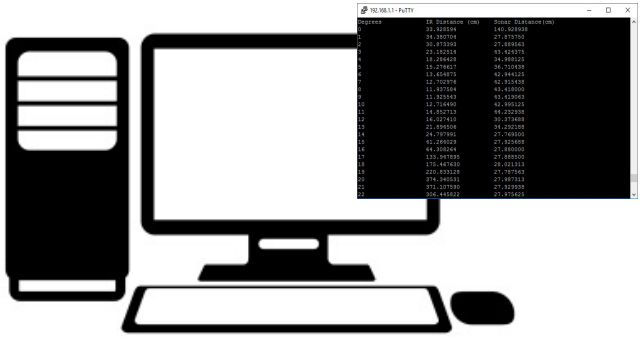


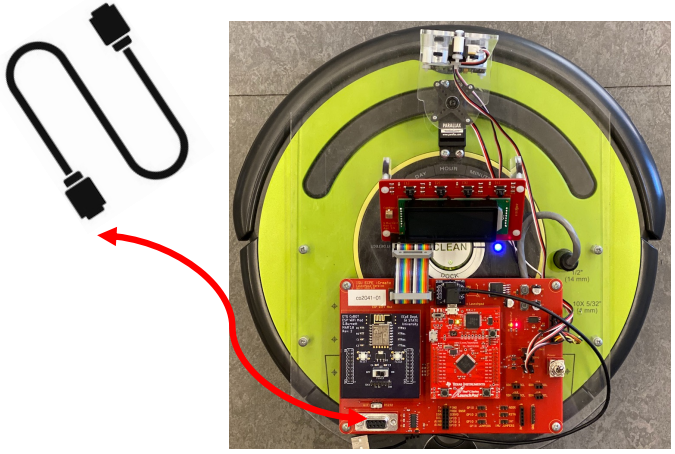
Figure 11.1. Synchronization Mechanisms

Note: "Fifo" in the flowchart is simply a buffer variable. It is different than the FIFO in the UART hardware interface.

Interactive tool and figure from VYES textbook

Background: UART Code in Labs 3 and 4

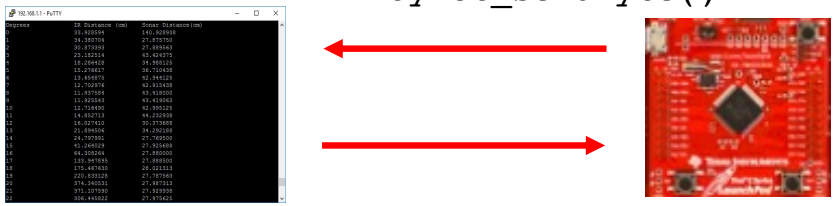




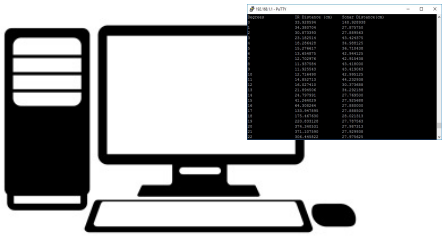
Code provided in pre-compiled library

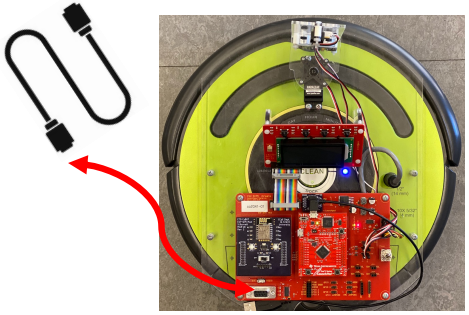
- `cyBot_uart.h`
- `libcybotUART.lib`

```
void cyBot_uart_init(void);
void cyBot_sendByte(char data);
int cyBot_getByte(void);
```



Background: UART Code in Lab 5 – Part 1



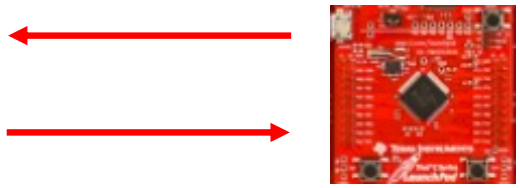


NEW pre-compiled library

- `cyBot_uart.h`
- `libcybotUART.lib`

```
void cyBot_uart_init(void);
void cyBot_uart_init_clean(void);
void cyBot_uart_init_PHJ_first_half(void);
void cyBot_uart_init_last_half(void);
void cyBot_sendByte(char data);
char cyBot_getByte_blocking(void);
```

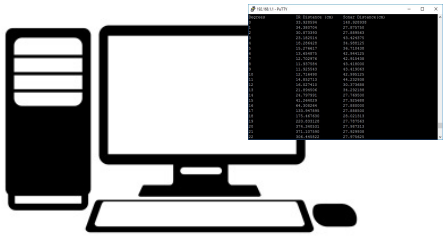
`cyBot_sendByte()`

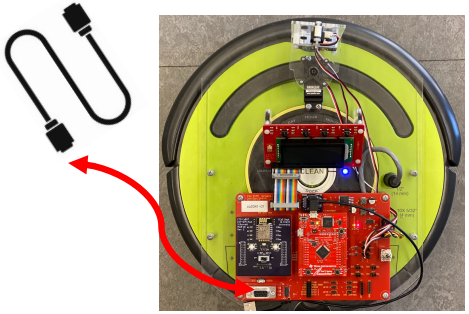


`cyBot_getByte_blocking()`

```
cyBot_uart_init_clean()
//YOUR OWN CODE FOR GPIO INIT
cyBot_uart_init_last_half()
```

Background: UART Code in Lab 5 – Part 2



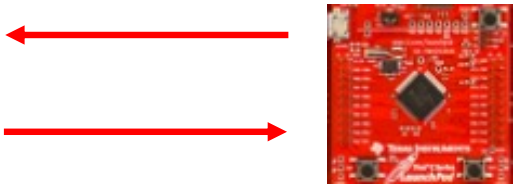


Your own code

- `uart.h`
- `uart.c`

```
void uart_init(void);
void uart_sendChar(char data);
char uart_receive(void);
void uart_sendStr(const char *data);
```

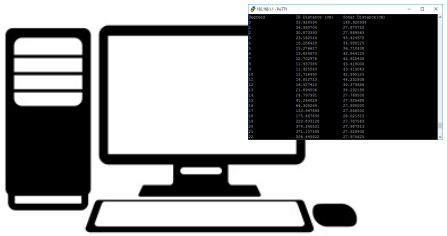
`uart_sendChar()`

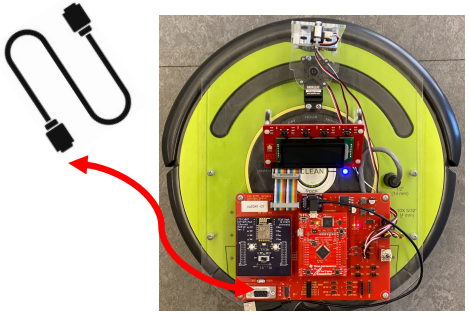


`uart_init ()`

`uart_receive()`

UART Code in Lab 6 – Part 1

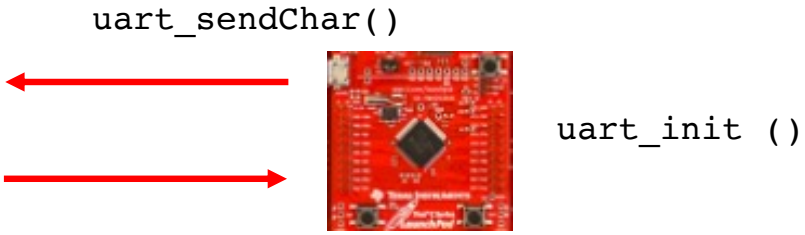




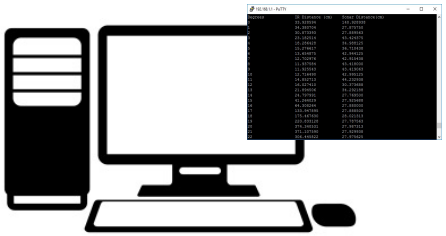
Your own code

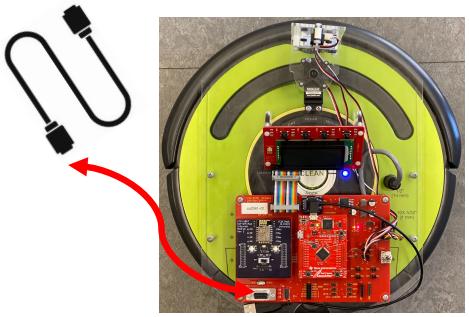
- `uart.h`
- `uart.c`

```
void uart_init(void);
void uart_sendChar(char data);
char uart_receive(void);
void uart_sendStr(const char *data);
int uart_receive_nonblocking(char *data);
```



UART Code in Lab 6 – Part 2





Your own code

- `uart-interrupt.h`
- `uart-interrupt.c`

```
void uart_interrupt_init(void);
void uart_sendChar(char data);
char uart_receive(void);
void uart_sendStr(const char *data);
void UART1_Handler(void);
```